Applicant : Shackleford et al.
Patent No. : n/a
Issued : n/a
Serial No. : 09/977,978
Filed : 10/17/2001
Page : 9

Attorney's Docket No.: 10019023-1

REMARKS

The Applicant appreciates the Examiner's careful consideration of the previous response and providing newly cited art as a result. However, the Applicant respectfully submits as indicated below, that aspects of the present invention as claimed remain patentable over these cited references as well.

Claims: Claims 1–6 were rejected under 35 U.S.C § 102(b) as being anticipated by Gutowitz (U.S. Patent 5,365,589).

However, the Examiner has failed to establish the prima facie case as each and every element of independent claim 1 and dependant claims 2-6 are not taught by the Gutowitz patent. See Richardson v. Suzuki Motor Co., 868 F.2d 1226, 1236, 9 U.S.P.Q.2D (BNA) 1913, 1920 (Fed. Cir.), cert. denied, 493 U.S. 853, 107 L. Ed. 2d 112, 110 S. Ct. 154 (1989) (explaining that an invention is anticipated if every element of the claimed invention, including all claim limitations, is shown in a single prior art reference). See Jamesbury Corp. v. Litton Industrial Products, Inc., 756 F.2d 1556, 1560, 225 USPQ 253, 256 (Fed. Cir. 1985) (explaining that the identical invention must be shown in as complete detail as is contained in the patent claim). See Verdegaal Bros., Inc. v. Union Oil Co., 814 F.2d 628, 631, 2 U.S.P.Q.2D (BNA) 1051, 1053 (Fed. Cir. 1987) (explaining that a prior art reference anticipates a claim only if the reference

discloses, either expressly or inherently, every limitation of the claim). See Kloster Speedsteel AB v. Crucible, Inc., 793 F.2d 1565, 1571, 230 U.S.P.Q. (BNA) 81, 84 (Fed. Cir. 1986) ("Absence from the reference of any claimed element negates anticipation.")

Gutowitz concerns encryption and decryption software that performs its operations over many cycles (Abstract). This is clearly identified in the specification and figures as well as in the claims (Col. 13, lines 47-68 of Gutowitz and FIG. 3; Claim 1). Specifically, claim 1 indicates "applying said at least one of said current-key dynamical systems over a selected number of iteration cycles to produce from said current state a new state of said at least one of said current-key dynamical systems". This makes sense since Gutowitz clearly states that they have merely specified particular computer-automata (CA) for encrypting and decrypting according to their particular scheme (Col. 9, lines 1-10). Namely, Gutowitz states, "The preferred embodiments of this invention use true cellular automata".

Indeed, Gutowitz also provides that encryption and decryption shall take place in parallel using special customized hardware structured to perform the functions of the CA (Col. 14, lines 32-44; FIG. 4 and Claim 14). Specifically, Claim 14 states "second data processing means connected to the first and second memory arrays for operating on data stored in said second memory array in accordance with the values in the first memory array to derive a new state of

the selected dynamical system". Essentially, Gutowitz indicates that to make their CA run in parallel you must use specialized hardware configured in a particular manner they specify in the various areas of the specification.

Both the general iterative and parallel approach described in Gutowitz is well know to those skilled in the art and presents nothing new. A single threaded software application implementing Gutowitz must operate in an iterative manner because the output from first computer automata must first be calculated before it can then feed a subsequent CA cell. Gutowitz mentions "simulating" the CA cells in software by literally performing the exact functions of each CA cell connected in sequence (Col. 4, lines 12-25) and operating encryption/decryption at a rate of 20,000 bits/second. It is no surprise that Gutowitz then states the special purpose hardware of course speeds this up from 100 to 1M times faster because it simply implements the CA cells in parallel. No heuristics or special techniques are used in Gutowitz to speed up the software as it is only a benchmark and comparison for the hardware implementation to be made subsequently.

In summary, Gutowitz can only operate CA in parallel using hardware that exactly replicates the function of the CA while the software in Gutowitz must operate iteratively and not in parallel as it too is merely replicating exact function of the CA. Even if multithreaded

software were suggested (which it was not), it would be merely performing the same CA functions in parallel by using multiple CPUs or CPU cores as directed by a sophisticated compiler or assembler.

Unfortunately, Gutowitz clearly does not teach or suggest, "software emulation of a cellular automata based random number generator (CA-based RNG), comprising:

determining a set of emulation parameters for the CA-based RNG;

initializing the software emulation according to the emulation parameters;

storing state values from odd-numbered cells in a first software variable and state values from even-numbered cells in a second software variable wherein word operations on the first and second variables enable the simulation of the cells to occur in parallel when executed; and

outputting a random number having the state values stored in the first software variable and the second software variable" as recited in claim 1.

The Examiner failed to point out with particularity where, if at all, any or all of these limitations were described in Gutowitz. Applicant respectfully submits after many hours of reviewing Gutowitz, none of these limitations are to be found. For example, Gutowitz never even as much as mentions "storing state values from odd-numbered cells in a first software variable and state values from even-numbered cells in a second software variable wherein word

operations on the first and second variables enable the simulation of the cells to occur in parallel when execute". In fact, Gutowitz *requires* specially designed hardware to implement a CA to operate in parallel (Col. 14, lines 32-44; FIG. 4 and Claim 14) and thus teaches away from a purely software solution. Gutowitz software simulations in contrast clearly run iteratively and not in parallel (Col. 13, lines 47-68 of Gutowitz and FIG. 3; Claim 1).

Applicants respectfully submit if Examiner believes that Gutowitz teaches each and every element of Claim 1 that these limitations are pointed out with particularity. Otherwise, we would respectfully request that the Examiner withdraw the rejection of claim 1 for failing the "all elements rule" required by 35 U.S.C § 102(b) and the MPEP.

Further, Claims 2-6 are not only allowable on their own but also allowable by virtue of their dependency on independent claim 1.

The Examiner also rejected claims 7-29 under 35 U.S.C § 102(b) as being anticipated by Lyke (U.S. Patent 6,215,327). Once again, the Applicants respectfully submit that Lyke does not teach each and every limitation as recited in claim 7 and subsequent dependant claims thus this rejection must also be withdrawn.

Lyke concerns creating a field programmable gate array (FPGA) that may operate at a molecular level; that is, a molecular FPGA or MFPGA (Abstract and Summary of Invention –

Col 8, lines 60-70). It is Lyke's goal to make a programmable circuit that would be useful in nanotechnology type applications that may be needed in the near future. For the most part, Lyke describes creating a very complex arrangement of these cells in hardware to be used with a smaller amount of software that reprograms the functionality of the array (Col. 5, lines 14-20). More importantly, it is clear that Lyke is not describing a pure software invention or the creation of any type of compiler since the focus is on creating the FPGA or MFPGA as described and claimed.

Consequently, Lyke does not anticipate claim 7 which recites a method to "generate a software code emulating a cellular automata based random number generator (CA-based RNG)". In fact, Lyke makes no mention of emulating anything as specific as an RNG in software let alone in a MFPGA.

Further, Lyke does not describe "determining RNG parameters" as recited in claim 7. Because Lyke is not concerned specifically with an RNG it makes sense that no RNG type solutions in software or hardware are described or discussed.

Lyke does also not discuss "determining one or more programming language templates" or "determining functional definition of the CA-based RNG" or determining initialization routines for the CA-based RNG" or even "determining simulation routines for the CA-based

RNG" as recited in claim 7. The Examiner has failed to point out with particularity where if at all Lyke mentions any or all of these items as required by 35 U.S.C § 102(b).

At most, Lyke mentions CA as an expression of complexity by analogy (Col. 11, lines 30-70) but this does not appear to rise to describe each and every element of claim 7 as required by the "all elements rule". Of course, it also follows that Lyke does not describe "determining simulation results destination routines for the CA-based RNG" and "outputting code for the CA-based RNG" as recited in claim 7 since it makes no mention of RNG and the like in any detail. Lyke only mentions a pseudo-random number generator in the paper by Tsalides et al as one example and well-known use of CA as well as another by Chatopadhyay et a. Specifically, Lyke mentions these papers to point out that they do not teach what is described and the focus of Lyke: the spatially unraveling of CA in hardware (Col. 21, lines 65-67 and Col. 22, lines 33-35).

In summary, Lyke is not concerned with creating software systems to emulate CA but hardware FPGA and MFPGA to perform CA functions more flexibly. Lyke is not concerned with simulating or emulating CAs but actually implementing them directly in hardware vis a vis FPGA and MFPGA type hardware. Clearly, these do not teach each and every limitation of claim 7 and thus Lyke does not anticipate claim 7 under 35 U.S.C § 102(b).

Applicant : Shackleford et al.
Patent No. : n/a
Issued : n/a
Serial No. : 09/977,978
Filed : 10/17/2001
Page : 16

Attorney's Docket No.: 10019023-1

Likewise, claims 8-29 are allowable independently as well as based upon their dependency on claim 7.

Claims 1-29 remain in condition for allowance in light of both Gutowitz and Lyke. Applicant respectfully requests reconsideration of the rejections presented and their withdrawal in view of their remarks provided hereinabove.

Applicants have made a diligent effort to place the claims in condition for allowance. However, should there remain unresolved issues that require adverse action, it is respectfully requested that the Examiner telephone Leland Wiesner, Applicants' Attorney at (650) 853-1113 so that such issues may be resolved as expeditiously as possible.

For these reasons, and in view of the above amendments, this application is now considered to be in condition for allowance and such action is earnestly solicited.

10/28/2005
Date

Respectfully Submitted,

Leland Wiesner
Attorney/Agent for Applicant(s)
Reg. No. 39424

Leland Wiesner, Attorney
Wiesner & Associates
366 Cambridge Avenue
Palo Alto, California 94306
Tel. (650) 853-1113